

Context reduction	Unoptimized K	K
Configuration and initialization $Config ::= Int \mid \llbracket Pgm \rrbracket \mid \llbracket Pgm, State \rrbracket$ $\llbracket p \rrbracket \rightarrow \llbracket p, \emptyset \rrbracket$ $\llbracket v, \sigma \rrbracket \rightarrow v$ $Cxt ::= \square \mid \llbracket Cxt, State \rrbracket$	$ConfigItem ::= k(K) \mid state(State)$ $Config ::= Int \mid \llbracket K \rrbracket \mid \llbracket Set[ConfigItem] \rrbracket$ $\llbracket p \rrbracket = \llbracket k(p) \ state(\emptyset) \rrbracket$ $\llbracket \langle k(v) \rangle \rrbracket = v$	– same – – same – – same – – same –
Variable lookup $\llbracket c, \sigma \rrbracket[x] \rightarrow \llbracket c, \sigma \rrbracket[\sigma[x]]$	$k(\frac{x}{\sigma[x]} \ state(\sigma))$	– same –
Operators $Cxt ::= \dots \mid Cxt + AExp$ $Cxt ::= \dots \mid AExp + Cxt$ $i_1 + i_2 \rightarrow i_1 +_{Int} i_2$ $Cxt ::= \dots \mid Cxt \leq AExp$ $Cxt ::= \dots \mid Int \leq Cxt$ $i_1 \leq i_2 \rightarrow i_1 \leq_{Int} i_2$ $Cxt ::= \dots \mid \text{not } Cxt$ $\text{not } bool \rightarrow \text{not}_{Bool} bool$ $Cxt ::= \dots \mid Cxt \text{ and } BExp$ $\text{true and } b \rightarrow b$ $\text{false and } b \rightarrow \text{false}$	$a_1 + a_2 \rightleftharpoons a_1 \curvearrowright \square + a_2$ $a_1 + a_2 \rightleftharpoons a_2 \curvearrowright a_1 + \square$ $i_1 + i_2 \rightarrow i_1 +_{Int} i_2$ $a_1 \leq a_2 \rightleftharpoons a_1 \curvearrowright \square \leq a_2$ $i \leq a_2 \rightleftharpoons a_2 \curvearrowright i \leq \square$ $i_1 \leq i_2 \rightarrow i_1 \leq_{Int} i_2$ $\text{not } b \rightleftharpoons b \curvearrowright \text{not } \square$ $\text{not } bool \rightarrow \text{not}_{Bool} bool$ $b_1 \text{ and } b_2 \rightleftharpoons b_1 \curvearrowright \square \text{ and } b_2$ $\text{true and } b \rightarrow b$ $\text{false and } b \rightarrow \text{false}$	$_ + _ \ [strict\ ext_ +_{Int} _]$ $_ \leq _ \ [seqstrict\ ext_ \leq_{Int} _]$ $\text{not_} \ [strict\ ext\ \text{not}_{Bool} _]$ $_ \text{ and_} \ [strict(1)]$ – same – – same –
Statements $Cxt ::= \dots \mid Var := Cxt$ $\llbracket c, \sigma \rrbracket[x := v] \rightarrow \llbracket c, \sigma[v/x] \rrbracket[\text{skip}]$ $Cxt ::= \dots \mid Cxt; Stmt$ $\text{skip}; s \rightarrow s$ $Cxt ::= \dots \mid \text{if } Cxt \ Stmt \ Stmt$ $\text{if true } s_1 \ s_2 \rightarrow s_1$ $\text{if false } s_1 \ s_2 \rightarrow s_2$ $\text{while } b \ s \rightarrow \text{if } b \ (s; \text{while } b \ s) \ \text{skip}$ $C ::= \dots \mid \text{halt } Cxt$ $\llbracket c, \sigma \rrbracket[\text{halt } i] \rightarrow i$	$x := a \rightleftharpoons a \curvearrowright x := \square$ $k(\frac{x := v}{\sigma[v/x]} \ state(\frac{\sigma}{\sigma[v/x]}))$ \cdot $s; k \rightleftharpoons s \curvearrowright k$ $\text{skip} = \cdot$ $\text{if } b \ s_1 \ s_2 \rightleftharpoons b \curvearrowright \text{if } \square \ s_1 \ s_2$ $\text{if true } s_1 \ s_2 \rightarrow s_1$ $\text{if false } s_1 \ s_2 \rightarrow s_2$ $k(\text{while } b \ s) = k(\text{if } b \ (s; \text{while } b \ s) \ \cdot)$ $\text{halt } a \rightleftharpoons a \curvearrowright \text{halt } \square$ $k(\text{halt } i) = k(i)$	$_ := _ \ [strict(2)]$ – same – – same – – same – $\text{if_} \ [strict(1)]$ – same – – same – – same – $\text{halt_} \ [strict]$ – same –
Programs $Cxt ::= \dots \mid Cxt; AExp$ $\text{skip}; a \rightarrow a$		

Figure 1: Definition of a simple language using context reduction, unoptimized K and K